WHAT IS CLAIMED IS:

Possible restriction

1. A method of parsing an input text segment according to a left-corner chart parsing technique which populates a chart according to a plurality of productions, the method comprising:

receiving the input text segment;

generating proposed incomplete edges, with mothers and predictions, based on the set of productions and based on the input text segment;

for each proposed incomplete edge:

performing a bottom-up left-corner check on the prediction of the proposed incomplete edge; and

if the bottom-up left-corner check on the prediction of the proposed incomplete edge is successful, performing a top-down left-corner check on the mother of the proposed incomplete edge, otherwise, not adding the proposed incomplete edge to the chart.

2. The method of claim 1 and further comprising:

if the proposed incomplete edge passes both the bottom-up left-corner check on the prediction of the proposed incomplete edge and the top-down left-corner check on the mother of the proposed incomplete edge, populating the chart with the proposed incomplete edge.

3. The method of claim 1 wherein performing the bottom-up left-corner check on the prediction of the proposed incomplete edge comprises:

for every complete edge of the form $\langle X,k,j \rangle$ in the chart and every production with X as its left-most daughter, of the form $A \rightarrow XY\alpha$, determining whether the j+1st terminal input symbol, $a_{j+1,}$ is a left corner of Y, wherein $\langle X,k,j \rangle$ represents a terminal or nonterminal which begins at a kth position in the input text segment and ends at the jth position in the input text segment, Y represents a terminal or nonterminal, α represents a sequence of terminals or nonterminals, and A represents a category which is the mother of the production.

4. The method of claim 3 wherein determining whether the j+1st terminal input symbol, $a_{j+1,}$ is a left corner of Y, comprises:

examining a left-corner table to determine whether it contains a pair of values including the j+1st terminal input and the left corner of prediction Y.

5. The method of claim 4 wherein, if the left-corner table includes the pair, concluding that the bottom-up left-corner check on the prediction is satisfied, and if not, concluding that the bottom-up left-corner check on the prediction is not satisfied.

6. The method of claim 1 wherein performing the top-down left-corner check on the mother of the proposed incomplete edge comprises:

for every complete edge of the form $\langle X,k,j \rangle$ in the chart and every production with X as its left-most daughter, of the form $A \rightarrow XY\alpha$, determining whether there is a B which is an element of P_k, such that A is a left corner of B, wherein B represents a category and P_k represents a set of predictions of incomplete edges in the chart ending at position k in the input text segment, wherein the prediction of an incomplete edge is a first as yet unmatched symbol of the incomplete edge.

7. The method of claim 6 wherein determining whether there is a B which is an element of P_k, such that A is a left-corner of B, comprises:

examining a left-corner table to determine whether it indicates that A is a left corner of B.

8. The method of claim 7 wherein, if the left-corner table indicates that *A* is a left corner of *B*, adding the proposed incomplete edge to the chart, otherwise, not adding the proposed incomplete edge to the chart.

9. A left-corner chart parser configured to populate a chart according to productions by performing the steps of:

receiving the input text segment;

generating proposed incomplete edges, with mothers and predictions, based on the set of productions and based on the input text segment;

for each proposed incomplete edge:

performing a bottom-up left-corner check on the prediction of the proposed incomplete edge; and

if the bottom-up left-corner check on the prediction of the proposed incomplete edge is successful, performing a top-down left-corner check on the mother of the proposed incomplete edge, otherwise, not adding the proposed incomplete edge to the chart.

10. A computer readable medium containing instructions which, when executed, cause the computer to parse an input text segment according to a left-corner chart

parsing method which populates a chart according to a plurality of productions, the method comprising:

receiving the input text segment;

generating proposed incomplete edges, with mothers and predictions, based on the set of productions and based on the input text segment;

for each proposed incomplete edge:

performing a bottom-up left-corner check on the prediction of the proposed incomplete edge; and

if the bottom-up left-corner check on the prediction of the proposed incomplete edge is successful, performing a top-down left-corner check on the mother of the proposed incomplete edge, otherwise, not adding the proposed incomplete edge to the chart.

11. The computer readable medium of claim 10 and further comprising:

if the proposed incomplete edge passes both the bottom-up left-corner check on the prediction of the proposed incomplete edge and the top-down left-corner check on the mother of the proposed incomplete edge, populating the chart with the proposed incomplete edge.

12. The computer readable medium of claim 10 wherein performing the bottom-up left-corner check on the prediction of the proposed incomplete edge comprises:

for every complete edge of the form $\langle X,k,j \rangle$ in the chart and every production with X as its left-most daughter, of the form $A \rightarrow XY\alpha$, determining whether the j+1st terminal input symbol, $a_{j+1,}$ is a left corner of Y, wherein $\langle X,k,j \rangle$ represents a terminal or nonterminal which begins at a kth position in the input text segment and ends at the jth position in the input text segment, Y represents a terminal or nonterminal, α represents a sequence of terminals or nonterminals, and A represents a category which is the mother of the production.

13. The computer readable medium of claim 12 wherein determining whether the j+1st terminal input symbol, $a_{j+1,}$ is a left corner of Y, comprises:

examining a left-corner table to determine whether it contains a pair of values including the j+1st terminal input and the left corner of prediction Y.

14. The computer readable medium of claim 13 wherein, if the left-corner table includes the pair, concluding that the bottom-up left-corner check on the prediction

is satisfied, and if not, concluding that the bottom-up left-corner check on the prediction is not satisfied.

15. The computer readable medium of claim 10 wherein performing the top-down left-corner check on the mother of the proposed incomplete edge comprises:

for every complete edge of the form $\langle X,k,j \rangle$ in the chart and every production with X as its left-most daughter, of the form $A \rightarrow XY\alpha$, determining whether there is a B which is an element of P_k, such that A is a left corner of B, wherein B represents a category and P_k represents a set of predictions of incomplete edges in the chart ending at position k in the input text segment, wherein the prediction of an incomplete edge is a first as yet unmatched symbol of the incomplete edge.

16. The computer readable medium of claim 15 wherein determining whether there is a B which is an element of P_k, such that A is a left-corner of B, comprises:

examining a left-corner table to determine whether it indicates that A is a left corner of B.

17. The computer readable medium of claim 16 wherein, if the left-corner table indicates that A is a left corner of B, adding the proposed incomplete edge to the

chart, otherwise, not adding the proposed incomplete edge to the chart.

18. A method of indexing productions for use in a left-corner chart parser which parses input text containing input symbols, the method comprising:

indexing the productions first based on input symbols which are consistent with the productions.

19. The method of claim 18 wherein indexing comprises:

precomputing which of the productions are consistent with which of the input symbols.

20. The method of claim 19 wherein precomputing comprises:

precomputing, for each possible input symbol, which productions have a second daughter with that input symbol as a left corner.

21. The method of claim 20 wherein indexing, comprises:

generating a data structure that, for each of the possible input symbols, includes a discrimination tree just for productions having a second daughter with that input symbol as a left corner.

22. The method of claim 18 and further comprising:
indexing the productions next based on a leftmost daughter of the productions.

23. The method of claim 22 and further comprising:
indexing the productions next based on a mother of the productions.

24. The method of claim 23 and further comprising:
enumerating the productions based on remainder of the productions, other than the leftmost daughter and the mother.

25. A method of parsing input text using a left-corner chart parsing process, comprising:
receiving an input symbol in the input text;
accessing an input symbol index to obtain productions having the input symbol as a left corner of the second daughter; and
after obtaining the productions having the input symbol as a left corner of the second daughter, accessing other indices to the productions.

26. The method of claim 25 wherein the input symbol index comprises a portion of a discrimination tree for only the productions having a second daughter with the input symbol as a left corner of the

second daughter, and wherein accessing the index comprises:

traversing the discrimination tree.

27. The method of claim 25 wherein accessing other indices comprises:

accessing a left-most daughter index to obtain productions based on their left-most daughter.

28. The method of claim 27 wherein accessing other indices comprises:

accessing a mother index to obtain productions based on their mother.

29. The method of claim 28 and further comprising:

accessing a list containing a completion of productions that are obtained by accessing the left-most daughter index and the mother index.

30. A data structure indexing productions used in a left-corner chart parser which parses input text, the data structure comprising:

a first index portion indexing the productions first based on input symbols which are consistent with the productions.

31. The data structure of claim 30 wherein the first index portion indexes productions by input symbol based on which productions have the input symbol as a left corner of the second daughter.

32. The data structure of claim 31 and further comprising:

a second index portion indexing the productions based on a left-most daughter of the productions.

33. The data structure of claim 32 and further comprising:

a third index portion indexing the productions based on a mother of the productions.

34. The data structure of claim 33 and further comprising:

a fourth portion enumerating the productions based on a remainder of the productions, other than the left-most daughter and the mother of the productions.

35 The data structure of claim 34 wherein the first, second, third and fourth portions comprise a discrimination tree implemented as a set of nested hash tables.

36. A method of transforming a grammar used in left-corner chart parsing, wherein the grammar includes a set of productions, each production having a mother, the method comprising:

applying a bottom-up prefix merging transformation regardless of the mother of the production; and

providing a transformed grammar.

37. The method of claim 36 wherein applying a bottom-up prefix merging transformation comprises:

identifying productions having similar symbols in similar positions on a right side of the productions; and

applying the bottom-up prefix merging transformation to the identified productions regardless of the mother of the identified productions.

38. The method of claim 37 wherein identifying productions comprises:

identifying productions having similar prefix symbols on the right side of the productions.

39. The method of claim 36 wherein applying a bottom-up prefix merging transformation comprises:

identifying productions of the form $A_1 \rightarrow \alpha\beta_1$, ..., $A_n \rightarrow \alpha\beta_n$, where α is a sequence of two or more symbols, and transforming the identified productions into transformed productions of the form $A' \to \alpha$, $A_1 \to A'\beta_1$, ..., $A_n \to A'\beta_n$, where A' is a new nonterminal symbol.

40. The method of claim 39 and further comprising:

repeating the steps of identifying and transforming until no further productions are identified.

41. A computer readable medium having stored thereon a data structure comprising a grammar used in left-corner chart parsing, the grammar including:

a set of productions having mothers, the set of productions being bottom-up prefix merged regardless of their mothers.

42. A computer readable medium including instructions readable by a computer which, when executed, transform a grammar used in left-corner chart parsing, the grammar including a set of productions, and each production having a mother, the transform comprising:

applying a bottom-up prefix merging transformation regardless of the mother of the production; and

providing a transformed grammar.

43. The computer readable medium of claim 42 wherein applying a bottom-up prefix merging transformation comprises:

identifying productions having similar symbols in similar positions on a right side of the productions; and

applying the bottom-up prefix merging transformation to the identified productions regardless of the mother of the identified productions.

44. The computer readable medium of claim 43 wherein identifying productions comprises:

identifying productions having similar prefix symbols on the right side of the productions.

45. The computer readable medium of claim 42 wherein applying a bottom-up prefix merging transformation comprises:

identifying productions of the form $A_1 \rightarrow \alpha\beta_1$, ..., $A_n \rightarrow \alpha\beta_n$, where α is a sequence of two or more symbols, and transforming the identified productions into transformed productions of the form $A' \rightarrow \alpha$, $A_1 \rightarrow A'\beta_1$, ..., $A_n \rightarrow A'\beta_n$, where A' is a new nonterminal symbol.

46. A method of flattening a grammar used in left-corner chart parsing, wherein the grammar includes productions, the method comprising:

eliminating single-option chain rules from the grammar to obtain a flattened grammar; and

output the flattened grammar.

47. The method of claim 46 and further comprising:

identifying single-option chain rules of the form $A \rightarrow X$, where A is a mother, and X is a single terminal or nonterminal daughter, to obtain identified productions.

48. The method of claim 47 wherein eliminating single-option chain rules from the grammar comprises:

omitting the identified productions from the grammar; and

substituting the daughter of the production for the mother of the production in remaining productions of the grammar.

49. A method of flattening a grammar used in left-corner chart parsing, wherein the grammar includes productions, the method comprising:

flattening the grammar based only on left-most daughters of the productions to obtain a flattened grammar; and

outputting the flattened grammar.

50. The method of claim 49 wherein flattening the grammar comprises:

for each nonterminal of the form A, determining whether A is a non-left-recursive category;

if so, determining whether A appears as a daughter of a production only if it is a left corner of the production; and

if so, flattening the grammar with respect to A.

51. The method of claim 50 wherein flattening the grammar with respect to A comprises:

for each production of the form $A \rightarrow X_1 \ldots X_n$, and each production of the form $B \rightarrow A\alpha$, adding $B \rightarrow X_1 \ldots X_n\alpha$ to the grammar; and

removing all productions containing A from the grammar.

52. The method of claim 50 and further comprising:

prior to flattening the grammar, determining whether there is a production which has A as a mother and at least one nonterminal as a daughter; and

if so, only then flattening the grammar with respect to A.

53. A computer readable medium having stored thereon instructions which, when executed, cause the computer to perform a method of flattening a grammar used in left-corner chart parsing, wherein

the grammar includes productions, the method comprising:

eliminating single-option chain rules from the grammar to obtain a flattened grammar; and

outputting the flattened grammar.

54. The method of claim 53 and further comprising:

identifying single-option chain rules of the form *A* → *X*, where *A* is a mother, and *X* is a single terminal or nonterminal daughter, to obtain identified productions.

55. The method of claim 54 wherein eliminating single-option chain rules from the grammar comprises:

omitting the identified productions from the grammar; and

substituting the daughter of the production for the mother of the production in remaining productions of the grammar.

56. A computer readable medium having stored thereon instructions which, when executed, course the computer to perform a method of flattening a grammar used in left-corner chart parsing, wherein the grammar includes productions, the method comprising:

flattening the grammar based only on left-most daughters of the productions to obtain a flattened grammar; and

outputting the flattened grammar.

57. The method of claim 56 wherein flattening the grammar comprises:

for each nonterminal of the form *A*, determining whether *A* is a non-left-recursive category;

if so, determining whether *A* appears as a daughter of a production only if it is a left corner of the production; and

if so, flattening the grammar with respect to *A*.

58. The method of claim 57 wherein flattening the grammar with respect to *A* comprises:

for each production of the form $A \rightarrow X_1 \ldots X_n$, and each production of the form $B \rightarrow A\alpha$, adding $B \rightarrow X_1 \ldots X_n\alpha$ to the grammar; and

removing all productions containing *A* from the grammar.

59. The method of claim 57 and further comprising:

prior to flattening the grammar, determining whether there is a production which has *A* as a mother and at least one nonterminal as a daughter; and

if so, only then flattening the grammar with respect to *A*.

60. A computer readable medium having stored thereon a data structure comprising a grammar used in left-corner chart parsing, the grammar comprising:

a set of productions having single-option chain rules removed therefrom.

61. A computer readable medium having stored thereon a data structure comprising a grammar used in left-corner chart parsing, the grammar comprising:

a set of flattened productions, flattened based substantially only on left-most daughters of the productions.

62. A method of assembling one or more analyses, based on a derived edge, of an input text parsed using a chart parser, the method comprising:

accessing a pointer associated with the derived edge which points to a first data structure containing a complete edge category and starting position in the input text for a first complete edge used in deriving the derived edge; and

assembling the analysis based on the complete edge category and starting address pointed to.

63. The method of claim 62 and further comprising:

prior to assembling the analysis, determining an ending position of the first complete edge.

64. The method of claim 63 and further comprising:

computing an incomplete edge used, with the first complete edge, to derive the derived edge.

65. The method of claim 64 and further comprising: prior to assembling the analysis, determining whether any additional complete edges are to be obtained.

66. The method of claim 65 wherein determining whether any additional complete edges are to be obtained comprises:

determining whether a starting position in the most recently computed incomplete edge is the same as a complete edge it was derived from.

66 ~~67~~. The method of claim 62 wherein the pointer associated with the derived edge points to additional data structures containing complete edge categories and starting positions in the input text for additional complete edges used in deriving the derived edge, and wherein assembling comprises assembling additional analyses based on information in the additional data structures.

67 ~~68~~. A method of storing edges completed during parsing of an input text, the method comprising:

storing in a data structure, only mother categories and starting positions of -

complete edges that were used in a final step of a derivation of a derived edge.

68 ~~69~~. The method of claim 68 and further comprising: storing a pointer from the derived non-initial edge to the data structure containing mother categories and starting positions.

69 ~~70~~. A computer readable medium including a data structure stored thereon, the data structure used in identifying complete edges obtained by performing a parse of an input text to obtain a derived edge, the data structure comprising one or more pairs of data portions including:

a first data portion containing only a category of a mother of a complete edge used to derive the derived edge; and

a second data portion containing only a starting position in the input text of the complete edge used to derive the derived edge, the data structure being formed regardless of an ending position of the complete edge.

70 ~~71~~. The computer readable medium of claim 70 wherein the data structure is attached to the derived edge.

71 ~~72~~. A computer readable medium including a data structure stored thereon, the data structure used in identifying complete edges obtained by performing a

chart parse of an input text to obtain a derived edge, the data structure comprising one or more pairs of data portions consisting essentially of:

a first data portion containing a category of a mother of a complete edge used to derive the derived edge; and

a second data portion containing a starting position in the input text of the complete edge used to derive the derived edge.

12

~~7~~3. A computer readable medium having stored thereon instructions which, when executed, cause the computer to perform a method of assembling an analysis, based on a derived edge, of an input text parsed using a chart parser, the method comprising:

accessing a pointer associated with the derived edge which points to a first data structure containing a complete edge category and starting position in the input text for a first complete edge used in deriving the derived edge; and

assembling the analysis based on the complete edge category and starting address pointed to.

13

~~7~~4. The computer readable medium of claim 73 and further comprising:

prior to assembling the analysis, determining an ending position of the first complete edge.

74 ~~75~~. The computer readable medium of claim 73 and further comprising:

computing an incomplete edge used, with the first complete edge, to derive the derived edge.

75 ~~76~~. The computer readable medium of claim 75 and further comprising:

prior to assembling the analysis, determining whether any additional complete edges are to be obtained.

76 ~~77~~. The method of claim 76 wherein determining whether any additional complete edges are to be obtained comprises:

determining whether a starting position in the most recently computed incomplete edge is the same as a complete edge it was derived from.

77 ~~78~~. The computer readable medium of claim 73 wherein the pointer associated with the derived edge points to additional data structures containing complete edge categories and starting positions in the input text for additional complete edges used in deriving the derived edge, and wherein assembling comprises assembling additional analyses based on information in the additional data structures.

78 ~~79~~. A computer readable medium having stored thereon instructions which, when executed cause the computer to perform a method of storing edges completed during parsing of an input text, the method comprising:

storing in a data structure, only mother categories and starting positions of complete edges that were used in a final step of a derivation of the derived edge.

79 ~~80~~. The computer readable medium of claim 79 and further comprising:

storing a pointer from the derived non-initial edge to the data structure containing mother categories and starting positions.